Modleamiento y predicción de lluvias usando Edge Computing para el entorno colombiano

Irene Arroyo Delgado, Oscar Carrillo, Frédéric Le Mouël

Rainfall Modeling and Prediction using Edge Computing for the Colombian Environment

Irene Arroyo Delgado[1][0000−0003−3014−2395], Oscar Carrillo[2][0000−0001−5081−1774], and Frédéric Le Mouël[3][0000−0002−7323−4057]

[1] Industrial Engineering, Universidad Politécnica de Madrid, Spain
irene.ad97@gmail.com
[2] Univ Lyon, CPE Lyon, INSA Lyon, CITI, F-69621 Villeurbanne, France
oscar.carrillo@cpe.fr
[3] Univ Lyon, INSA Lyon, CITI, F-69621 Villeurbanne, France
frederic.le-mouel@insa-lyon.fr

Abstract. Nowadays the number of devices connected to internet which offer the possibility to collect data is increasing. The interconnectivity of these new sensors favors the creation of sustainable cities, in which the optimization of resources is based on the collected data. These sensors are also a big source of information for forecasting future values.
In this work we present an Edge Computing approach for the analysis and forecasting of rainfall data that is later validated on the CITI Laboratory Youpi Platform. To this end, we built a container image with the necessary tools and libraries to use the time series prediction models SARIMA and Prophet on ARMv7 architectures. A Raspberry Pi 3 node was chosen to evaluate performance on an Edge Computing device.
Colombia was chosen due to its tropical location and its variant geography which present a wide range of historical rainfall data. The data we used to train our models consisted on the historical mesures from sensors deployed in Colombia by the "Instituto de Hidrología, Meteorología y estudios Ambientales de Colombia - IDEAM". In first place, we selected Bucaramanga to study its data sensors and to define the well-suited parameters for SARIMA and Prophet trend models. The comparison between them presented a high degree of similarity, offering a good prediction of dry and wet seasons. Thereafter, the SARIMA and Prophet model of Bucaramanga were used to observe its adaptability to the cities of Bogotá and Medellín, getting a successful outcome at seasonal predictions.
After this estimation of the SARIMA parameters and its analysis offline, a container image was created to simplify and speed up the models implementation in the devices for predicting the two years monthly rainfall for Bucaramanga, Bogotá and Medellín. The container is available for armv7 architectures, that is usually used for IoT nodes on the Youpi platform.
The proposed model allows to create a network of sensors, with distributed analysis capacity, that improve the prevention of flood or drought emergencies in Smart Cities on Colombia, helping to manage resources for agriculture or prevent catastrophes.

Keywords: Data analytic · Docker · Internet of Things (IoT) · SARIMA · time series prediction · Prophet · Raspberry Pi · Edge Computing.

1 Introducción

Hoy en día los conceptos de IoT [2] y Smart City [4] están adquiriendo un papel fundamental en la revolución de la Industria 4.0. Según la empresa internacional de investigación y consultoría de tecnologías de la información Gartner, se espera que para 2020 haya más de 20 mil millones de dispositivos de IoT conectados [6].

Un alto porcentaje de dispositivos IoT se encuentran integrados en las llamadas Smart City, su fin es estudiar el comportamiento de las ciudades para establecer medidas que permitan hacer un uso eficiente de los recursos como el transporte e infraestructuras. La utilización de machine learning y análisis de datos [13] ha sido introducida en las ciudades inteligentes, permitiendo la mejora la calidad de vida e impulsando la economía, al favorecer el uso de predicciones para los modelos de producción y consumo sostenible. En ellas se deben afrontar nuevos retos de innovación con el fin aumentar la seguridad y privacidad, además de la calidad de los datos que se analizan[8].

1.1 Proposición

A partir de la previsión de aumento de la población se ha acentuado la necesidad de economizar los recursos de los que se dispone, preservando el medio ambiente para las futuras generaciones. Para llevar a cabo una adecuada gestión de los recursos, una de las herramientas claves es la previsión, que se basa en la toma de medidas con el fin de predecir y prepararse para mitigar las catástrofes medioambientales. Una referencia para la prevención de riesgos es la ciudad de Praga, con su estudio de simulación de inundaciones con el fin de establecer protocolos de emergencia para las situaciones más críticas[12]. En este contexto nace el presente trabajo donde se realiza la predicción de la precipitación de la región de Bucaramanga en Colombia, mediante modelos de predicción de series temporales SARIMA y Prophet en nodos de IoT. Para ambos modelos de predicción, se estudia su portabilidad para la ciudad de Bogotá y Medellín. La finalidad, es crear una red de sensores puedan actuar de manera autónoma en las estaciones de meteorología y prevengan de comportamientos anómalos basados en la predicción que se realice a partir de los datos históricos. El contenido de la aplicación que se ejecuta en las Raspberry Pi 3 para la realización de las predicciones es:

1. Análisis de los datos.
2. Predicción.
3. Validación de la predicción.

2 Metodología

Los métodos utilizados para la estimación de los modelos son SARIMA y Prophet que a continuación se detalla su ajuste.

2.1 SARIMA

SARIMA "Seasonal AutoRegressive Integrated Moving Averages" es un conocido modelo estadístico para analizar series de datos y predecir futuros valores únicamente con la dependencia que existe entre los datos históricos [5,11].

El modelo $SARIMA(p,d,q)(P,D,Q,m)$ se constituye de 7 parámetros, los primeros tres (p,d,q) representan la parte regular del modelo, mientras que (P,D,Q,m) representan la parte estacional.

P, p: término de la componente auto regresiva (AR).
D, d: término de la componente diferencial(I).
Q, q: término de la componente de media móvil (MA).
m: periodicidad de la serie.

Para determinar los parámetros, se debe seguir la metodología Box-Jenkins:

1. Identificación y estimación del modelo: Comprobación de la estacionaridad, identificación si procede de estacionalidad y determinación de p y q a partir de las gráficas de autocorrelación y autocorrelación parcial.
2. Determinación de parámetros con algoritmos de cálculo: A través del valor AIC ("Akaike Information Criterion") junto con el error cuadrático medio (ECM), se obtienen unos parámetros de mayor precisión.
3. Verificación: Se analizan los coeficientes, la bondad de ajuste y se comprueba que los residuos sigan un proceso de ruido blanco.
4. Predicción: Una vez que todas las condiciones anteriores se cumplen, el modelo esta listo para realizar la predicción, bajo la hipótesis que los valores futuros están relacionados con el pasado. Se realiza una predicción de 24 meses.

2.2 Prophet

Facebook publicó en 2017 una herramienta llamada Prophet. Es una librería básica de código abierto que permite hacer modelos y predicciones de series temporales. Se basa en un modelo de regresión aditivo o también llamado "curve fitting" que se representa por la siguiente ecuación.

$$y(t) = g(t) + s(t) + h(t) + \epsilon_t \tag{1}$$

$g(t)$: función de tendencia de crecimiento logarítmico o linear para modelos con cambios no periódicos en la serie.

$s(t)$: función de cambios periódicos y estacionales.
$h(t)$: efecto de las vacaciones y eventos.
ϵ_t: término de errores no modelizados.

En el modelo Prophet, se configura la estacionalidad anual, los valores de saturación, el porcentaje de valores donde aplicar los puntos de cambio de tendencia y el intervalo de confianza. Mientras que los parámetros internos del modelo se autoajustan para cada serie. Con el objetivo de la validación del modelo, Prophet incorpora para el diagnóstico la validación cruzada ("cross validation") y las métricas de rendimiento ("metrics performance") [14,9].

3 Implementación

Los modelos se han determinado offline a partir de los datos de Bucaramanga y posteriormente, se han exportado a un nodo de IoT para la predicción de datos futuros.

3.1 Análisis de los Datos

Para la realización de los modelos, se necesitaba en primer lugar, una base de datos fiable y segura. Para dicha aplicación, se analizó la precipitación acumulada mensual. La fuente de datos fue el Instituto de Hidrología, Meteorología y estudios Ambientales- IDEAM [7]. Los sensores proporcionaban la precipitación acumulada diaria en mm de las estaciones de la región de Bucaramanga, a los que se les debió pasar un filtro para descartar los datos fuera de los límites naturales. El filtro máximo fue seleccionado en base a cuál había sido la precipitación máxima diaria de los últimos años en la zona donde se sitúa la estación, mientras que el filtro mínimo fue cero mm por obviedad.

3.2 Modelo SARIMA para Bucaramanga

Para la elección de los siete parámetros del modelo SARIMA, fue necesario seguir los pasos anteriormente mencionados con el fin de obtener el modelo para Bucaramanga. En primer lugar, se comprobó la estacionaridad, para ello se usó el método de Dickey-Fuller. Al aplicar la función a la serie anteriormente tratada, se obtuvo que el método pasaba para todos los intervalos de confianza estudiados, pero la media y varianza no seguín una tendencia estacionaria. Por lo que se decidió aplicar diferencia de orden uno. La diferencia de orden uno para el parámetro regular aportó una media y varianza estacionarias, por lo tanto, se adoptó para el modelo d=1 y D=0. Posteriormente, con el fin de determinar manualmente una aproximación de los parámetros del modelo, se analizó la ACF y PACF de la serie. Estas funciones presentaban una forma de abanico que se completaba en un periodo aproximadamente 12, m=12. Por lo que, se observó que poseía componentes estacionales y un ciclo anual. Al ser la modelización de series meteorológicas tan compleja, los parámetros restantes se obtuvieron

con el método AIC. Se procedió a encontrar el modelo con menor AIC y menor ECM. Con esta finalidad, se realizó un "grid search" donde se obtuvieron los tres modelos con parámetros entre [0,3] con menor AIC. De los cuales, se seleccionó como mejor modelo el que menor error disponga. Para la validación del modelo, se determinanó los parámetros de ajuste entrenando el modelo con el 90% de los datos y posteriormente verificando con el 10% restante. A partir de la grid search se obtuvo que el modelo elegido es SARIMA (2, 1, 3) (2, 0, 3, 12) al ser el que menor error cuadrático medio presentaba. Los modelos SARIMA ofrecen la posibilidad de estudiar el peso e importancia de sus parámetros. Para este modelo, se estudió que ar.S.L24, el parámetro correspondiente a P de orden 2, tenía un valor P>z muy alto y un coeficiente pequeño, es decir tenía poco peso y una importancia despreciable en la predicción por lo que se podía eliminar simplificando el modelo. El modelo definitivo fue SARIMA (2, 1, 3) (1, 0, 3, 12). El cual fue validado al comprobar que sus residuos seguían una distribución de ruido blanco, es decir los residuos seguían un proceso de distribución aleatorio, Figura 1, con una correcta aproximación de la predicción para el 10% de los datos usados en el previo entrenamiento y un buen ajuste a los datos reales de la serie.

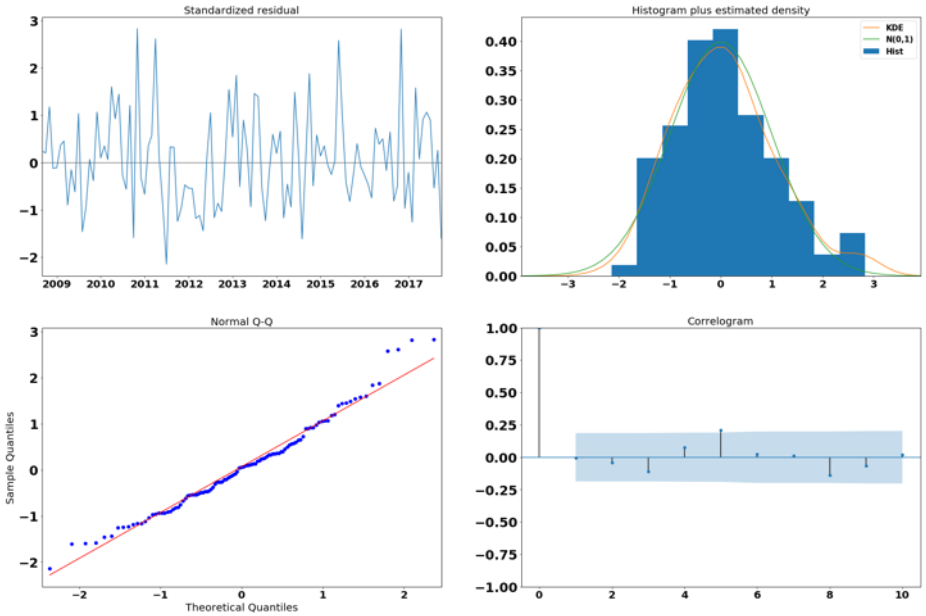

Fig. 1. Residuos del modelo SARIMA

Con el modelo ya establecido se realiza la predicción para 24 meses, Figura 2, dos años, con un intervalo de confianza del 95%. Esta predicción sigue los estándares marcando las temporadas de lluvias y secas.

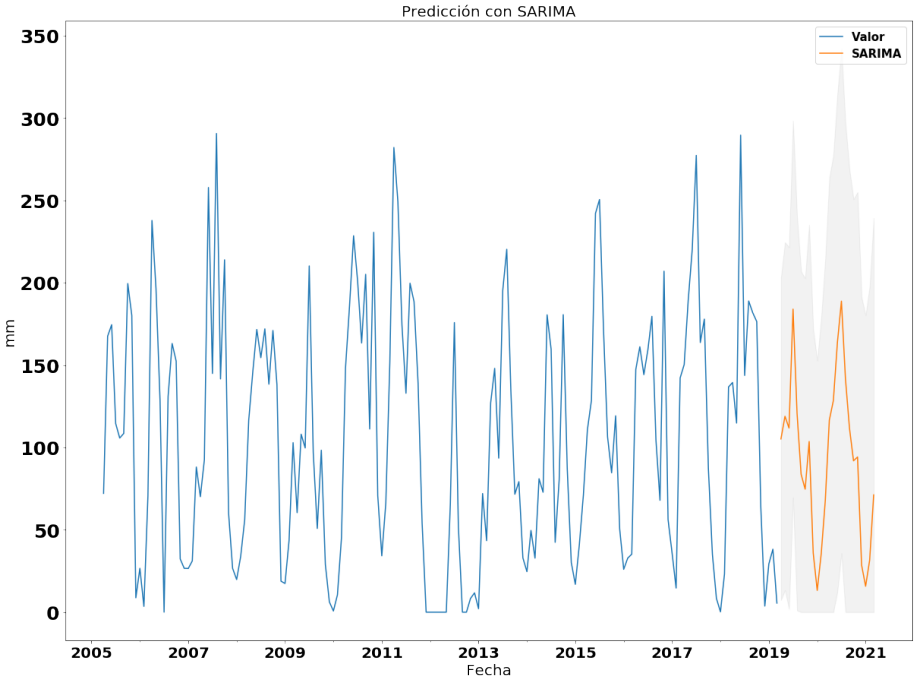

Fig. 2. Predicción para 24 meses en Bucaramanga con SARIMA

3.3 Modelo Prophet para Bucaramanga

En Prophet para Python, se analizan los puntos de cambio de tendencia en el 80% de los datos, para una mayor exactitud en el presente trabajo se aumentó hasta el 90% dejando espacio para la proyección de la tendencia, pero analizándose un mayor número de puntos. Por otro lado, se aumentó la flexibilidad de la tendencia, añadiendo más prioridad a los puntos de cambio "changepoint" a un 0.7 debido a que en dicho modelo es necesario ajustarse a las tendencias de los años posteriores. Además, se eliminaron los datos atípicos y en su lugar, el modelo estimaría que valores corresponderían según la tendencia en ese intervalo de tiempo. Esta propiedad se utilizó con el fin de que no se tuviesen en cuenta los datos de 2006-07, 2011-12 - 2012-06, 2012-09 y 2012-10 que estarían definidos a 0 mm, para la estación de Paramo del Almorzadero, Bucaramanga. Otro aspecto a destacar ha sido la fijación de la saturación mínima a 0, debido a que no es físicamente posible obtener lluvia negativa. La predicción para 24 meses fue efectuada con el modelo Prophet siguiendo una tendencia realista, cumpliendo los objetivos con un intervalo de confianza del 95% como se puede corroborar en la Figura 3 .

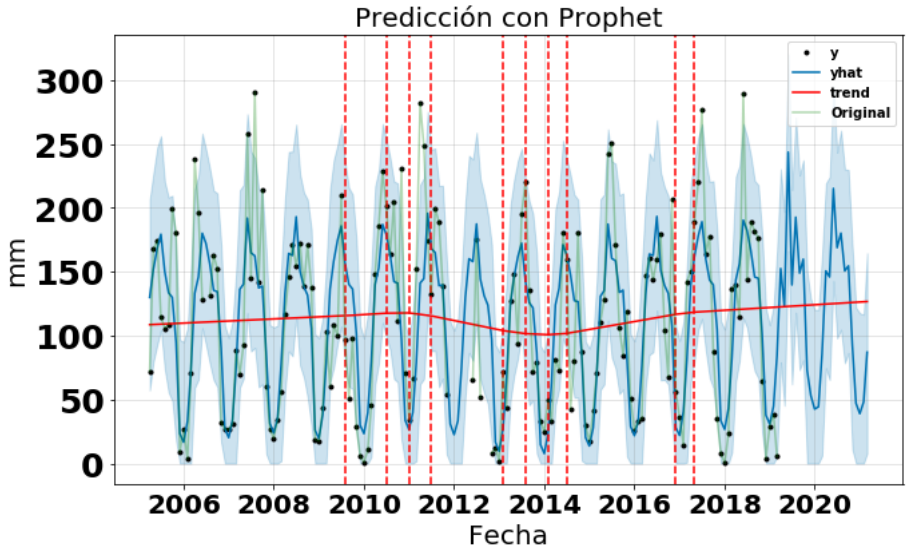

Fig. 3. Predicción para 24 meses en Bucaramanga con Prophet

4 Modelo en Nodos IoT

A fin de ejecutar los modelos creados en Jupyter notebook es necesario tener instaladas las librerías específicas para cada proceso, además del propio Jupyter

notebook y Python en la Raspberry Pi 3. Este no es un dispositivo IoT, sin embargo, cuenta con una arquitectura propia de una amplia cantidad de ellos, por lo que se utilizan para realizar simulaciones de comportamiento. El proceso de descarga de cada paquete es lento al tener la Raspberry Pi una conexión Ethernet más lenta que un ordenador personal al compartir el bus USB de la placa para realizar la conexión. De manera que se agilice el tiempo de descarga y su portabilidad, se creó una imagen Docker con todo lo necesario, con el fin automatizar su implementación.

Se implementó la imagen Docker para arm/v7 desde el Escritorio Docker a partir de compilación cruzada. Esta funcionalidad fue anunciada el 24 de abril de 2019 por lo que es una herramienta todavía en versión beta. Docker Desktop o Escritorio Docker, ha incorporado esta funcionalidad debido a la gran expansión de los dispositivos de IoT que presentan estas arquitecturas ARM [10].

5 Resultados y Conclusiones

En primer lugar, se realizó un estudio de los diferentes sensores del Instituto de Hidrología, Meteorología y estudios Ambientales- IDEAM para la ciudad de Bucaramanga. Se estudiaron catorce sensores de los cuales, al aplicar el filtro de corrección de datos y representarlo gráficamente, se observó que en todos los sensores menos en uno había grandes intervalos de datos atípicos o intervalos de datos escuetos, por lo que se llegó a la conclusión de que *la única estación correcta para la realización de predicción de series temporales era Paramo del Almorzadero*, que se encuentra situada a 46 km de Bucaramanga.

Por otra parte, al realizar un estudio similar posteriormente para Bogotá y Medellín se pudo observar nuevamente estos intervalos prolongados de datos atípicos y siniestros en las estaciones. De esta manera se afirmó que *las estaciones de IDEAM en las ciudades de Bucaramanga, Bogotá y Medellín presentaban un elevado número de sensores con datos incorrectos.*

Posteriormente, se estudió la serie temporal de Páramo del Almorzadero para determinar un modelo SARIMA y Prophet que se ajustase correctamente y proporcionase una predicción fiable para la ciudad de Bucaramanga. Para el modelo SARIMA, se utilizó el método AIC combinado con el error cuadrático medio, obteniendo como resultado que el mejor modelo era SARIMA (2, 1, 3) (2, 0, 3, 12). Tras un análisis de sus coeficientes se observó que se podía reducir el orden de AR(P) sin alterar el resto de los parámetros, por lo que se obtuvo como modelo final $SARIMA(2,1,3)(1,0,3,12)$.

Para el modelo con Prophet, los parámetros son autoajustables, por lo que solo se identificaron las zonas con datos atípicos.

Al comparar las predicciones de la Figura 4 , se observó que presentaban los mismos cambios de estaciones, época lluviosa entre abril-mayo y septiembre-octubre, y temporada seca en los intervalos restantes, siendo más relevante enero donde la precipitación es mínima. Por lo tanto, comparadas con el clima en esta región y sus estaciones, la predicción presenta un estudio realista de los periodos estivales y con respecto a la predicción de valores, ambas presentan un elevado

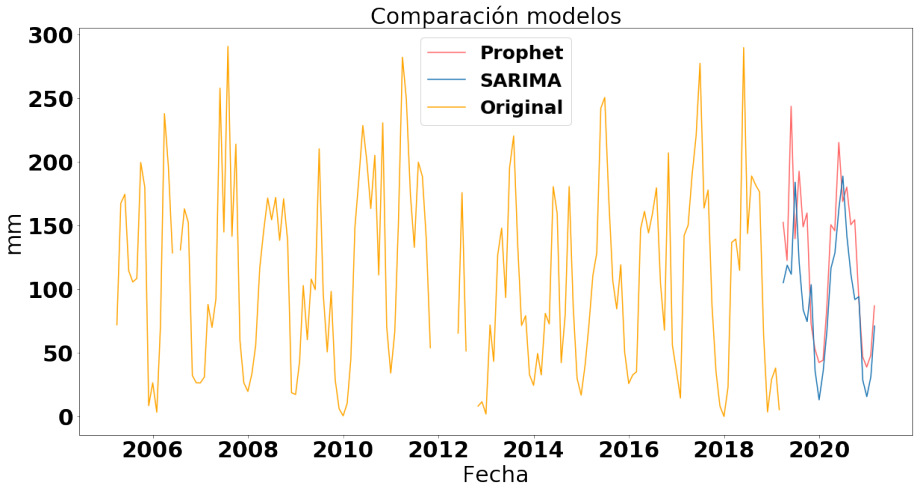

Fig. 4. Comparación modelos SARIMA y Prophet para Bucaramanga

valor aproximado entre ellas. Se concluye, que ambos modelos representan una predicción fiable, bien si a número de mm de precipitación acumulada mensual no es totalmente exacto, al incluir un nivel de confianza del 95%, se utilizará para estudiar las distintas temporadas y próximas tendencias con gran exactitud.

A continuación, se buscaba comparar como ajustan los modelos anteriormente seleccionados a otras series temporales, para comprobar si los modelos que se han seleccionado eran generalizables para otras ciudades con meteorología parecida. Se estudió para las ciudades de Bogotá y Medellín.

En el ajuste para Bogotá, se estudió el sensor de Páramo de Chingaza, situado a 38 km de Bogotá. La predicción de las estaciones de la Figura 5 coincide

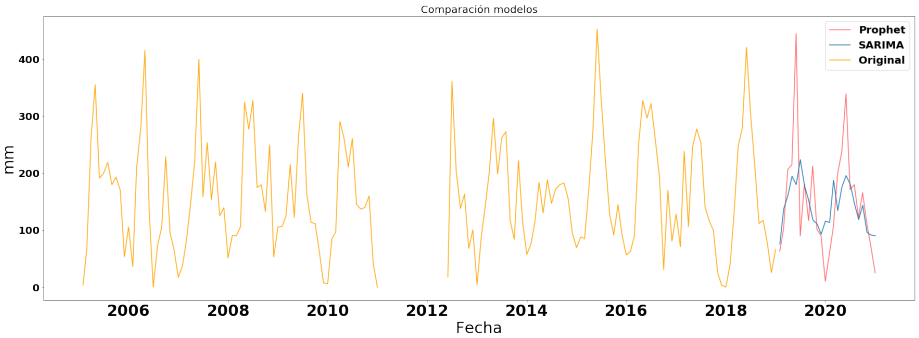

Fig. 5. Comparación modelos SARIMA y Prophet para Bogotá

con las de esta región, bien si a valor numérico para el caso de SARIMA es inferior al de Prophet en la temporada húmeda y mayor en la temporada seca. Se concluye que los modelos se ajustan satisfactoriamente a nivel de estaciones y respecto a una mayor exactitud de la precipitación mensual hay que incluir el intervalo de confianza. En el ajuste de Medellín, se estudió el sensor de Aragón a 60 km de Medellín La predicción que se realizó en la Figura 6 sigue los estándares

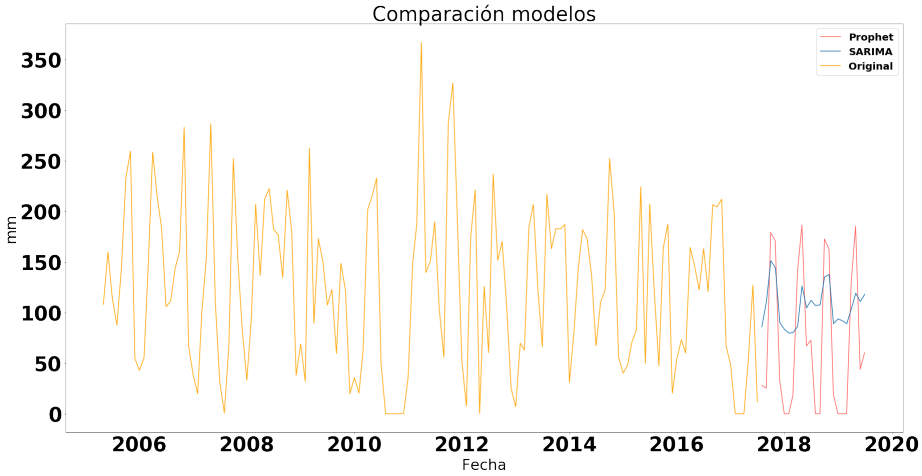

Fig. 6. Comparación modelos SARIMA y Prophet para Medellín

respecto a las estaciones húmedas y secas. Sin embargo, respecto a los valores de precipitación mensual el modelo SARIMA presenta unos niveles mínimos elevados. Se concluye que ambos modelos representan correctamente las temporadas de precipitación, pero a nivel de exactitud Prophet es más adecuado. Con el fin de exportar los modelos en Jupyter Notebook a los dispositivos IoT. Se creó una imagen Docker `dynamid/rain-forecast-iarroyo` con la cual se facilita y agiliza el proceso. Al comparar los tiempos de ejecución, el proceso en la Raspberry Pi 3 fue más lento que en el ordenador personal de un orden de 7.4 veces superior para SARIMA y 8.2 veces superior para Prophet. A pesar de su ejecución más costosa en tiempo, la Raspberry Pi 3 utilizada representa un dispositivo que soporta las herramientas necesarias para dichos modelos con un rendimiento aceptable.

Consecuentemente, se ha demostrado la factibilidad de ejecutar los modelos de predicción de lluvia en arquitecturas armv7 que hasta el momento no se había realizado debido a que los nodos no disponían de la capacidad de compilación de las herramientas de análisis de series de datos, que gracias a la compilación cruzada ha sido posible que se compilen en una arquitectura más eficiente.

Como síntesis final, se puede decir que se ha conseguido obtener los modelos de SARIMA y Prophet, para realizar una aplicación en Jupyter Notebook capaz de realizar predicciones de la precipitación exportables a una red de nodos IoT

con arquitectura armv7 gracias a la imagen Docker suponiendo un incremento en el tiempo respecto del ordenador personal debido a la arquitectura y componentes de la Raspberry. Los modelos elegidos para Bucaramanga son adecuados a su vez para Bogotá y Medellín para indicar las estaciones húmedas y secas, además de la tendencia para los próximos dos años [1].

Este trabajo supone un aporte en la transformación de las ciudades a Smart Cities con la capacidad de predecir y poder simular posibles escenarios de precipitación meteorológica y mitigar los efectos adversos de un cambio de tendencia gracias a herramientas de machine learning y data analytic. Con este proyecto, se abren las puertas a nuevas investigaciones para crear una plataforma de pruebas con una red de recogida de datos meteorológicos en tiempo real como la existente en Birmingham [3].

References

1. Arroyo Delgado, I.: Predicción de precipitaciones con dispositivos IoT mediante análisis de series temporales. B.S. Thesis, Universidad Politécnica de Madrid (2019)
2. Atzori, L., Iera, A., Morabito, G.: The Internet of Things: A survey. Computer Networks **54**(15), 2787–2805 (2010). https://doi.org/10.1016/j.comnet.2010.05.010, http://linkinghub.elsevier.com/retrieve/pii/S1389128610001568
3. Chapman, L., Muller, C.L., Young, D.T., Warren, E.L., Grimmond, C.S., Cai, X.M., Ferranti, E.J.: The Birmingham urban climate laboratory: An open meteorological test bed and challenges of the Smart city. Bulletin of the American Meteorological Society **96**(9), 1545–1560 (2015). https://doi.org/10.1175/BAMS-D-13-00193.1
4. Ching, T.Y., Ferreira, J.: Smart Cities: Concepts, Perceptions and Lessons for Planners, pp. 145–168. Springer International Publishing, Cham (2015). https://doi.org/10.1007/978-3-319-18368-8_8
5. de la Fuente Fernández, S.: Series Temporales: Modelo Arima. Universidad Autónoma de Madrid p. 53 (2016), http://www.estadistica.net/ECONOMETRIA/SERIES-TEMPORALES/modelo-arima.pdf
6. Hung, M.: Leading the IoT. Gartner insights on how to lead in a connected world. Gartner (2017), http://gartner.com/imagesrv/books/iot/iotEbook%5Fdigital.pdf
7. IDEAM: Instituto de Hidrología, Meteorología y estudios Ambientales, http://www.ideam.gov.co
8. Naphade, M., Banavar, G., Harrison, C., Paraszczak, J., Morris, R.: Smarter cities and their innovation challenges. Computer **44**(6), 32–39 (2011). https://doi.org/10.1109/MC.2011.187
9. Nishida, K.: An Introduction to Time Series Forecasting with Prophet in Exploratory (2017), https://blog.exploratory.io/an-introduction-to-time-series-forecasting-with-prophet-package-in-exploratory-129ed0c12112
10. Parko, A.: Building Multi-Arch Images for Arm and x86 with Docker Desktop (2019), https://engineering.docker.com/2019/04/multi-arch-images/
11. Peña, D.: Análisis de Series Temporales. Alianza (2005)
12. Rothkrantz, L.J.: Flood control of the smart city Prague. 2016 Smart Cities Symposium Prague, SCSP 2016 pp. 1–7 (2016). https://doi.org/10.1109/SCSP.2016.7501043

13. de Souza, J.T., de Francisco, A.C., Piekarski, C.M., do Prado, G.F.: Data mining and machine learning to promote smart cities: A systematic review from 2000 to 2018. Sustainability (Switzerland) **11**(4) (2019). https://doi.org/10.3390/su11041077
14. Taylor, S.J., Letham, B., Taylor, S.J., Letham, B.: Forecasting at Scale Forecasting at Scale. The American Statistician **72**(1), 37–45 (2018). https://doi.org/10.1080/00031305.2017.1380080